

Structures

```
1  #include <stdio.h>
2  #include <math.h>
3
4  struct line{
5      struct point{
6          float x;
7          float y;
8      }point1, point2;
9  };
10
11 float solveSlope(struct line line1){          // to solve for slope
12     float slope = (line1.point2.y - line1.point1.y) / (line1.point2.x - line1.point1.x);    // formula for slope
13     return slope;
14 }
15
16 struct point solveMidpoint(struct line line1){    // to solve for midpoint
17     struct point midpoint;
18     midpoint.x = (line1.point1.x + line1.point2.x) / 2;    // midpoint for x
19     midpoint.y = (line1.point1.y + line1.point2.y) / 2;    // midpoint for y
20     return midpoint;
21 }
22
23 float solveDistance(struct line line1){          // distance formula
24     float distance = sqrt(pow(line1.point2.x - line1.point1.x, 2) + pow(line1.point2.y - line1.point1.y, 2));
25     return distance;
26 }
27
28 void getSlopeInterceptForm(struct line line1){    // slope intercept form
29     float slope = solveSlope(line1);
30     float intercept = line1.point1.y - (slope * line1.point1.x);
31     printf("y = %.2fx + (%.5f)\n", slope, intercept);
32 }
33
```

```
33
34 int main(){
35     struct line line1 = {0};                // initialize line1 to zero
36     int choice;
37
38     do{
39         printf("\nEnter x and y for point1: ");    // ask user input for point1 coordinates
40         scanf("%f %f", &line1.point1.x, &line1.point1.y);
41
42         printf("Enter x and y for point2: ");    // ask user input for point2 coordinates
43         scanf("%f %f", &line1.point2.x, &line1.point2.y);
44
45         float slope = solveSlope(line1);    // calculate slope, midpoint, and distance
46         struct point midpoint = solveMidpoint(line1);
47         float distance = solveDistance(line1);
48
49         printf("Slope: %.2f\n", slope);    // to print slope, midpoint, distance, and slope-intercept form
50         printf("Midpoint: %.2f %.2f\n", midpoint.x, midpoint.y);
51         printf("Distance between 2 points: %.2f\n", distance);
52         getSlopeInterceptForm(line1);
53
54         printf("\nPress 1 to terminate the program or any other key to try again: ");    // ask if the user wants to continue or exit
55         scanf("%d", &choice);
56         printf("\n");
57     }while (choice != 1);
58
59     return 0;
60 }
61
62
```

Sample Output

```
Enter x and y for point1: 1 2
Enter x and y for point2: 3 4
Slope: 1.00
Midpoint: 2.00 3.00
Distance between 2 points: 2.83
y = 1.00x + (1.00000)

Press 1 to terminate the program or any other key to try again:
4
```

```
Enter x and y for point1: 12 32
Enter x and y for point2: 3 4
Slope: 3.11
Midpoint: 7.50 18.00
Distance between 2 points: 29.41
y = 3.11x + (-5.33334)

Press 1 to terminate the program or any other key to try again:
```

```
Enter x and y for point1: 32 4
Enter x and y for point2: 1 2
Slope: 0.06
Midpoint: 16.50 3.00
Distance between 2 points: 31.06
y = 0.06x + (1.93548)

Press 1 to terminate the program or any other key to try again: 1

C:\Users\Apong\Desktop>_
```

Github Link:

<https://github.com/kaisaaaaa/CMSC21/tree/b8efab85a9a02590d5d1344d344c701875a26ab3/Lecture13>